

Các hàm thành phần của lớp CSocket (MFC)

Lệnh này dùng để tạo socket

```
BOOL Create(  
  
    UINT nSocketPort = 0,  
  
    int nSocketType = SOCK_STREAM,  
  
    LPCTSTR lpszSocketAddress = NULL  
  
);
```

- nSocketPort: cổng trên host được dùng với Socket; nếu bằng 0 thì mặc định MFC sẽ tự chọn port.
- nSocketType là SOCK_STREAM (TCP) hay SOCK_DGRAM (UDP).
- lpszSocketAddress: địa chỉ IP của host dùng socket để truyền thông.
Ví dụ: 172.29.6.240.
- Giá trị trả về: khác 0 nếu thành công, ngoài ra thì bằng 0.

Lệnh dán địa chỉ cục bộ đến socket

```
BOOL Bind(  
  
    UINT nSocketPort,  
  
    LPCTSTR lpszSocketAddress = NULL  
  
);  
  
BOOL Bind (  
  
    const SOCKADDR* lpSockAddr,  
  
    int nSockAddrLen  
  
);
```

- nSocketPort: cổng xác định ứng dụng socket.

- lpszSocketAddress: địa chỉ mạng.
- lpSockAddr: trỏ đến cấu trúc SOCKADDR chứa địa chỉ cho socket này.
- nSockAddrLen: chiều dài của địa chỉ lpSockAddr được tính theo byte.
- Giá trị trả về: khác 0 nếu thành công, ngoài ra = 0.

```
struct sockaddr {  
    unsigned short sa\_family;  
    char sa\_data[14];  
};
```

- Đây là cấu trúc lưu trữ địa chỉ IP để máy gia nhập vào truyền thông socket windows.

Hàm giải phóng socket

Close()

Hàm thiết lập kết nối đến server

```
BOOL Connect(  
    LPCTSTR lpszHostAddress,  
    UINT nHostPort  
);  
  
BOOL Connect(  
    const SOCKADDR* lpSockAddr,  
    int nSockAddrLen );
```

- Gọi hàm này để thiết lập 1 kết nối đến 1 socket.
- lpszHostAddress: địa chỉ mạng của socket mà cần được kết nối đến, có thể là chuỗi số phân cách bởi dấu chấm, vd: 172.29.6.240.

- nHostPort: cổng xác định ứng dụng socket.
- lpSockAddr: trỏ đến cấu trúc SOCKADDR chứa địa chỉ của socket được kết nối.
- nSockAddrLen: chiều dài của địa chỉ trong lpSockAddr được tính theo byte.
- Giá trị trả về: khác 0 nếu thành công, thất bại = 0.

Hàm lắng nghe các yêu cầu kết nối

```
BOOL Listen(  
    int nConnectionBacklog = 5  
);
```

- Hàm này chỉ hỗ trợ cho socket dạng SOCK_STREAM.
- nConnectionBacklog: chiều dài tối đa mà hàng đợi kết nối chưa được chấp nhận có thể tăng. Phạm vi là từ 1 đến 5.
- Giá trị trả về: khác 0 = thành công, bằng 0 = thất bại.

Hàm chấp nhận một kết nối trên socket

```
virtual BOOL Accept(  
    CAsyncSocket& rConnectedSocket,  
    SOCKADDR* lpSockAddr = NULL,  
    int* lpSockAddrLen = NULL );
```

- Socket mới cho kết nối
- lpSockAddr trỏ đến cấu trúc SOCKADDR nhận địa chỉ mạng của socket kết nối (trên client).
- Nếu lpSockAddr hay lpSockAddrLen lấy giá trị mặc định NULL thì sẽ không có thông tin từ socket (trên client) được chấp nhận được trả về.

- lpSockAddrLen chứa chiều dài thực sự của lpSockAddr khi trả về theo byte.
- Giá trị trả về: khác 0 nếu thành công và bằng 0 nếu thất bại.

Hàm nhận dữ liệu từ socket (TCP)

```
virtual int Receive(  
    void* lpBuf,  
    int nBufLen,  
    int nFlags = 0  
);
```

- lpBuf: vùng đệm chứa dữ liệu.
- nBufLen: chiều dài của vùng đệm tính theo byte.
- nFlag: cách nhận dữ liệu, sử dụng giá trị mặc định là 0.
- Giá trị trả về là số byte nhận được, nếu socket đóng thì giá trị trả về là 0, ngoài ra giá trị sẽ trả về là SOCKET_ERROR.

Hàm gửi dữ liệu đến socket (TCP)

```
virtual int Send(  
    const void* lpBuf,  
    int nBufLen,  
    int nFlags = 0  
);
```

- lpBuf: vùng đệm chứa dữ liệu để truyền đi.
- nBufLen: chiều dài vùng đệm.
- nFlag: cách truyền dữ liệu, sử dụng giá trị mặc định là 0.

- Giá trị trả về là số ký tự được gửi, nếu thất bại giá trị trả về là SOCKET_ERROR.

Hàm nhận dữ liệu từ Socket (UDP)

```
int ReceiveFrom(  
    void* lpBuf,  
    int nBufLen,  
    CString& rSocketAddress,  
    UINT& rSocketPort,  
    int nFlags = 0  
);  
  
int ReceiveFrom(  
    void* lpBuf,  
    int nBufLen,  
    SOCKADDR* lpSockAddr,  
    int* lpSockAddrLen,  
    int nFlags = 0 );
```

- Hàm nhận 1 datagram và lưu địa chỉ trong rSocketAddress hay trong cấu trúc SOCKADDR từ 1 nguồn cụ thể.
- lpBuf: vùng đệm chứa dữ liệu nhận được.
- rSocketAddress: tham chiếu đến đối tượng CString nhận địa chỉ IP.
- rSocketPort: trỏ đến 1 biến UINT chứa port.
- lpSockAddr: trỏ đến cấu trúc chứa địa chỉ nguồn.
- lpSockAddrLen: chứa chiều dài của địa chỉ nguồn lưu trong lpSockAddr.
- nFlag: cụ thể cách gọi.

- Giá trị trả về là số byte nhận được, = 0 nếu socket bị đóng, = SOCKET_ERROR nếu có lỗi.

Hàm gửi dữ liệu đến socket (UDP)

```
int SendTo(  
    const void* lpBuf,  
    int nBufLen,  
    UINT nHostPort,  
    LPCTSTR lpszHostAddress = NULL,  
    int nFlags = 0  
);  
  
int SendTo(  
    const void* lpBuf,  
    int nBufLen,  
    const SOCKADDR* lpSockAddr,  
    int nSockAddrLen,  
    int nFlags = 0  
);
```

- lpBuf: vùng đệm chứa dữ liệu để truyền.
- nBufferLen: chiều dài của dữ liệu trong vùng đệm tính theo byte.
- nHostPort: port ứng dụng.
- lpszHostAddress: địa chỉ của máy được kết nối, vd:172.29.6.240.
- lpSockAddr: trỏ đến cấu trúc chứa địa chỉ của socket đích.
- lpSockAddrLen: chiều dài của địa chỉ lưu trong lpSockAddr tính theo byte.
- Trả về số ký tự được gửi, ngoài ra trả về SOCKET_ERROR khi có lỗi.

Hàm vô hiệu hoá luồng chức năng

```
BOOL ShutDown(  
  
    int nHow = sends  
  
);
```

- Hàm làm vô hiệu hóa send, receive hay cả hai.
- nHow = 0: disable receive
- nHow = 1: disable send
- nHow = 2: disable both
- Trả về giá trị khác 0 nếu thành công, và trả về 0 nếu lời gọi hàm thất bại.

— Hết —